

- **Interface UART** : L'application devra proposer un shell afin de communiquer voir reconfigurer sans recompilation des variables d'environnement du firmware embarqué. L'application devra proposer une interface de communication (cf. shell) implémentant un jeu de commande prédéfini dans le firmware. Jeu de commande à supporter :
 - **eth** : passe en mode ethernet et stop le mode wifi. Le serveur TCP (ou UDP ou les deux) embarqué côté SoC duplique l'interface UART vers l'ordinateur de supervision (client TCP ou UDP ou les deux).
 - **wifi** : passe en mode wifi et stop le mode eth. La seule communication réseau IP active est la communication WIFI vers la tablette
 - **ip** : modification de l'adresse IP pour le mode ethernet. 4 valeurs entière séparées par un point , par exemple : 192.168.94.1
 - **netmask** : modification du masque de sous réseau pour le mode ethernet. 4 valeurs entière séparées par un point , par exemple : 255.255.255.0
 - **config** : affiche les principales informations IP du mode courant (ethernet ou wifi). Par exemple : adresse IP, masque de sous réseau, ports TCP et UDP, mode de fonctionnement (ad hoc ou infrastructure) ... ne pas hésiter à rajouter des champs si ils vous semblent importants.
 - **list** : affiche tous les modules boost connectés ainsi les principales informations techniques concernant chaque module. Chaque module apparaîtra sous le nom suivant boost0, boost1 ... (le premier connecté étant le 0, puis le 1 ... etc). Nous afficheront également les valeurs moyennes de la tension de sortie, la tension d'entrée, le courant d'entrée, la puissance en entrée et le courant estimé de sortie (rendement unitaire). Affiche "none" si aucun module n'est connecté.
 - **help** : retourne le jeu de commandes supportées ainsi qu'une courte aide

L'interface proposée par le SoC devra absolument respecter l'affichage suivant (exemple de d'implémentation du jeu de commandes) :

cf. page suivante